10. FILTERING AND DATA PREPROCESSING FOR TIME SERIES ANALYSIS

William J. Randel

National Center for Atmospheric Research
Boulder, Colorado

10.1 Filtering Time Series

10.1.1 Introduction

The analysis of time series in physical sciences usually involves procedures such as differentiating, integrating, smoothing, extrapolating, or removal of noise. These all involve linear transformations of the original data, and the application of a linear transformation to a time series may be viewed as some sort of digital filter applied to that time series. In this section some general aspects of applying filters to time series are discussed. The behavior of a filter is often characterized in terms of its *frequency response function;* the motivation behind this concept is discussed, and the frequency response function is calculated for several examples. Next, the technique of designing a digital filter for a desired frequency response is discussed, including a simple FORTRAN subroutine for calculation of digital filter weights. Finally, the method of filtering time series by direct Fourier analysis–resynthesis is discussed. The fine details of filtering time series can be a rather complex subject, and several textbooks can be found that discuss these details (the book *Digital Filters* by R. W. Hamming [1] is a very readable and useful reference). The objective of this chapter is to give a brief introduction to some of the more practical aspects of using digital filters.

In practice, a digital filter is applied to a time series $u(t)$ by forming weighted linear combinations of successive subsets of the time series; letting $c(k)$ denote the weights, this produces a new "filtered" time series $y(t)$:

$$y(t) = \sum_{k=-M}^{M} c(k) \cdot u(t - k) \qquad (10.1)$$

This process is termed a *convolution* of the data $u(t)$ with the filter coefficients $c(k)$. The total number of filter coefficients here is $(2M + 1)$. Note

that the new or filtered time series $y(t)$ is shorter on each end than the original time series by M points.

A familiar example of this procedure is the smoothing of data by application of a "running 3-point average" or a "1-1-1 moving average" filter. The smoothed or filtered data is given by an average of three successive input values:

$$y(t) = [u(t + 1) + u(t) + u(t - 1)]/3$$

This is represented by (10.1) with $M = 1$ and $c(k) = 1/3$. This is the simplest type of filter to consider, called a *nonrecursive* filter, because it uses only the original time series as input data. Filters may also be considered that use prior calculated values of the output, e.g.,

$$y(t) = \sum_{k=-M}^{M} c(k) \cdot u(t - k) + \sum_{l=1}^{L} d(l) \cdot y(t - l)$$

These are termed *recursive* filters. For a given filter length M, recursive filters have better frequency response characteristics than nonrecursive filters, but they require more computational expense. Furthermore, their analysis is somewhat more complex, and the focus here is on nonrecursive filters.

10.1.2 Frequency Response of a Filter

Digital filters represent linear transformations of time series. In order to understand specifically what occurs in the transformation, it is useful to consider the process in the frequency domain. The frequency response of a filter indicates the transformation that occurs for each frequency component of the input and output time series. To transform to the frequency domain, begin by considering the (complex) finite Fourier series expansion of the time series $y(t)$ $(t = 0, 1, \ldots, N - 1)$:

$$\tilde{y}(\omega_j) = \frac{1}{N} \sum_{t=0}^{N-1} y(t) \cdot e^{-i\omega_j t} \tag{10.2}$$

The discrete angular frequencies ω_j are given by $\omega_j = (2\pi/N) \cdot j$, with $j = -N/2 + 1, \ldots, -1, 0, 1, \ldots, N/2$. For a real function $y(t)$, the positive and negative frequency coefficients are complex conjugates, i.e., $\tilde{y}(\omega_j) = \tilde{y} * (\omega_{-j})$. The inverse transform is

$$y(t) = \sum_{j=-N/2+1}^{N/2} \tilde{y}(\omega_j) \cdot e^{i\omega_j t} \tag{10.3}$$

The total variance of the time series $y(t)$ may be equated to the sum of the squared harmonic coefficients according to

$$\text{Variance } [y(t)] = \frac{1}{N} \sum_{t=0}^{N-1} [y(t) - \bar{y}]^2$$

$$= \sum_{j=1}^{N/2} 2 \cdot |\bar{y}(\omega_j)|^2 \tag{10.4}$$

where $\bar{y}$ is the average value of $y(t)$. A plot of $2 \cdot |\bar{y}(\omega_j)|^2$ versus ω_j is termed a *periodogram;* this quantity measures the contribution of oscillations with frequency near ω_j to the overall variance of the time series $y(t)$. Such a diagram is similar to the *sample spectral power density*, although the quantity $2 \cdot |\bar{y}(\omega_j)|^2$ is not a good estimate of the "true" underlying spectral density, because it is biased and its uncertainty does not decrease as the sample size increases. These factors are discussed in more detail in Chapter 11, along with techniques for the proper and consistent estimation of the spectral power density.

Now consider the frequency transform of (10.1):

$$\bar{y}(\omega_j) = \sum_{t=0}^{N-1} y(t) \cdot e^{-i\omega_j t}$$

$$= \sum_{t=0}^{N-1} \left[\sum_{k=-M}^{M} c(k) \cdot u(t - k) \right] e^{-i\omega_j t} \tag{10.5}$$

$$= \sum_{k=-M}^{M} c(k) \cdot e^{-i\omega_j k} \cdot \sum_{t=0}^{N-1} u(t - k) \cdot e^{-i\omega_j (t-k)}$$

$$\equiv \bar{c}(\omega_j) \cdot \bar{u}(\omega_j)$$

This result shows that the frequency coefficients of the filtered data $\bar{y}(\omega_j)$ are equal to the frequency coefficients of the original time series $\bar{u}(\omega_j)$, multiplied by the frequency transform of the filter coefficients $\bar{c}(\omega_j)$. The frequency transform $\bar{c}(\omega_j)$ thus measures what the filter does to each frequency coefficient and is termed the *frequency response* of the filter. Because $\bar{c}(\omega_j)$ is in general a complex quantity, it is composed of both amplitude and phase components. In terms of spectral power densities (10.4),

$$|\bar{y}(\omega_j)|^2 = |\bar{c}(\omega_j)|^2 \cdot |\bar{u}(\omega_j)|^2 \tag{10.6}$$

i.e., the spectral power density of the filtered data at each frequency is $|\bar{c}(\omega_j)|^2$ times the power density of the input time series.

Examples of the frequency response of several filters are included here to get a feeling for its use and meaning. The frequency response is calculated directly from the discrete transform equation:

$$\tilde{c}(\omega_j) = \sum_{k=-M}^{M} c(k) \cdot e^{-i\omega_j k} \qquad (10.7)$$

For simple filters, the summations can be done directly; more complicated filters require computer calculations.

Example 1:   We examine the frequency response for a 1-1-1 moving average filter. This is a filter with coefficients

$$c(-1) = 1/3$$

$$c(0) = 1/3$$

$$c(1) = 1/3$$

Using (10.7) the frequency response is evaluated as

$$\tilde{c}(\omega_j) = \frac{1}{3}\left(e^{-i\omega_j} + 1 + e^{-i\omega_j}\right)$$

$$= \frac{1}{3}(1 + 2\cos\omega_j) \qquad (10.8)$$

This response is shown in Fig. 1 as a function of frequency and wavelength (measured in terms of grid spaces or sampling intervals). In this and the following figures the physical frequency ($f = \omega/2\pi$) is used as abscissa, with units of (1/unit time) or (1/grid spacing). This response function shows the fraction of wave amplitude at each frequency that is passed through this filter. The response is near 1.0 at very low frequencies, denoting that
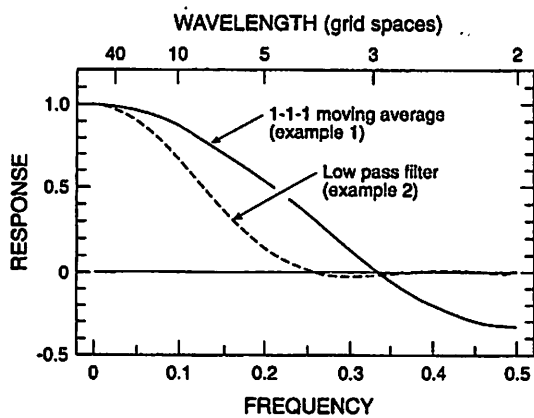

FIG. 1. Frequency response function calculated from (10.7) for the 1-1-1 moving average filter (Example 1) and low-pass filter (Example 2).

low-frequency oscillations (long wavelengths) are nearly unaffected by this filter. Conversely, higher frequency oscillations (shorter wavelengths) are selectively damped out; fluctuations near frequency 0.3 (3-grid space wavelength) are almost completely eliminated. This type of filter is termed a *low-pass filter*, because low frequencies are passed through while high frequencies are removed. This agrees with the intuitive knowledge that application of a 1-1-1 moving average removes high-frequency "noise" from a time series.

Note that the response function for this filter (10.8) is real valued, and this denotes that the phase of each frequency component remains unchanged by this filter. This is a general result for *symmetric* filters; i.e., ones with $c(-k) = c(k)$, and it is often a desirable property to be used in designing a filter. The negative response at high frequencies in Fig. 1 simply means that the input and output frequency coefficients are oppositely signed (i.e., that these frequency components in the filtered data will have opposite sign to those in the original time series).

To clearly show this transformation in the frequency domain, the sample spectral density function of a time series is examined before and after application of this filter. Figure 2 shows a time series of east–west winds near an altitude of 15 km, measured over Canton Island (near the equator in the central Pacific ocean) during 1960–1962. (This is a subset of a time series which was used to first discover the presence of a "40–50 day oscillation" in the winds, temperatures, and pressures over the Pacific Ocean [2]; this oscillation is now recognized as an important mode of atmospheric variability in the tropics.) Figure 3(a) shows a smoothed version of the spectral power density calculated from this time series, using the standard techniques discussed in Chapter 11. The spectral power density shows a strong peak in the frequency band .020–.025 days^{-1} (wave periods of 40–50
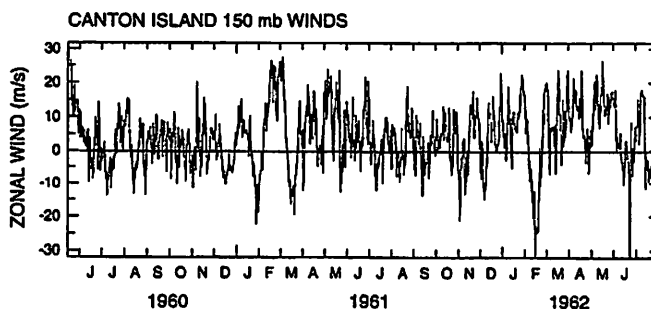

FIG. 2. Time series of east–west wind speed (m/s) near an altitude of 15 km, measured over Canton Island (in the equatorial Pacific Ocean) during 1960–1962.
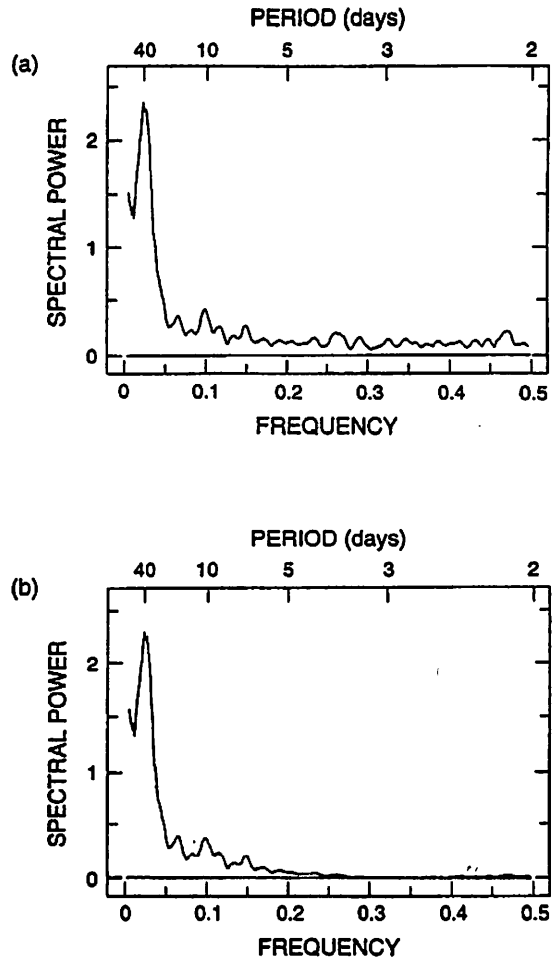
FIG. 3. Spectral power density calculated from the Canton Island time series in Fig. 2, (a) before and (b) after applications of a 1-1-1 moving average filter.

days), and near constant power at frequencies greater than $f = .1-.2$. Figure 3(b) shows the power spectral density for this time series after application of a 1-1-1 moving average filter. The spectral signature shows that the high frequency components ($f > 0.2$) have been selectively damped out. Note from (10.6) that the power spectral density for the filtered data is equal to that for the original data, multiplied by the square of the frequency response function (the square of the curve shown in Fig. 1). There is a small residual

of power between $f = 0.4$–0.5 in the filtered data in Fig. 3(b), due to the nonzero filter response seen for this frequency range in Fig. 1.

Example 2: We now look at a "better" low-pass filter. The nonzero frequency response of the 1-1-1 moving average filter near frequency $f = 0.5$ (see Fig. 1) means that some high-frequency components still remain in the filtered data. This may not be important for the time series in Fig. 1 because there is relatively little power at high frequencies; however, in general a more ideal low-pass filter would have a response function that approaches 0 at all frequencies above some high-frequency cutoff limit f_{high}. An example of such a filter is given by the filter coefficients

$$c(0) = 0.260000$$
$$c(1) = 0.217069$$
$$c(2) = 0.120215$$
$$c(3) = 0.034124$$
$$c(4) = 0.002333$$

and again the filter weights are symmetric $c(-k) = c(k)$. These filter weights were calculated by the subroutine included in Section 10.1.3 (see discussion later). The frequency response function for this filter is shown as the dashed curve in Fig. 1, as calculated by numerical summation of (10.7). The response is near 1.0 at low frequencies and near 0.0 at high frequencies, with a transition region near $f = 0.1$–0.2 (i.e., this filters most of the oscillations with periods shorter than 5–10 days). The width of the transition region (where the response goes from 1.0 to 0.0) is directly related to the number of coefficients chosen for this filter (in this case $M = 4$). The use of more filter coefficients will result in a sharper transition region, but at the cost of losing more data at the beginning and end of the filtered time series. In practice, the user must balance the choice of sharper frequency cutoff versus longer filter length for the specific time series at hand. In this example relatively few coefficients have been used, with the result that the transition region is broad. The result of this filter applied to the Canton Island time series is shown in Fig. 4, clearly showing the smoothing effect of a low-pass filter; note that four data points have been lost from the beginning and end of the filtered data.

The simple computer subroutine used to generate these low-pass filter coefficients will also generate coefficients for *high-pass filters* (which pass high frequencies and remove low-frequency components) and *bandpass filters* (which pass frequencies only over a specified frequency band). An example of the latter is included later (Example 4).
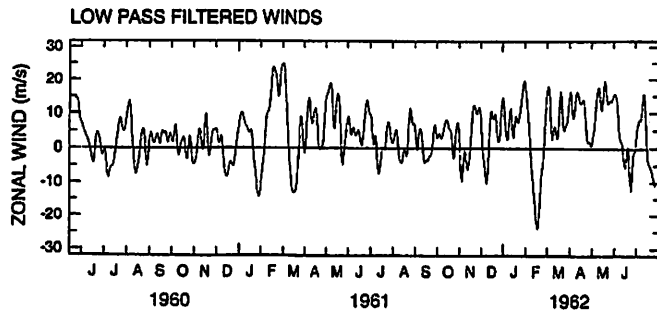
LOW PASS FILTERED WINDS

FIG. 4. Time series of Canton Island winds (Fig. 2), after application of the low-pass filter of Example 2.

Example 3: Finite difference approximations to derivatives are common calculations made on time series. The finite difference operators used to approximate derivatives are digital filters; the associated frequency response functions indicate how accurate the finite approximations are. Consider first the common centered difference operator

$$y(t) = [u(t + 1) - u(t - 1)]/2$$

This is a filter with coefficients

$$c(-1) = -1/2$$

$$c(0) = 0$$

$$c(1) = 1/2$$

The frequency response for this filter is given by (10.7):

$$\tilde{c}(\omega_j) = \frac{1}{2}(-e^{-i\omega_j} + e^{i\omega_j})$$

$$= i\sin\omega_j$$

Now for each spectral component $y(t) = e^{i\omega_j t}$, the "true" derivative is given by $y'(t) = i\omega_j y(t)$. The ratio of the calculated estimate of the continuous derivative to the true value is thus

$$\frac{\text{calculated}}{\text{true}} = \frac{\sin \omega_j}{\omega_j} \tag{10.9}$$

This calculated–true ratio was not explicitly considered in Examples 1 and 2, because there the ratio of the output to the input is the natural comparison. However, for derivatives (or integrals) the calculated result should be com-

pared to the ideal result; in both cases the ratio is viewed as a transfer or frequency response function.

The response curve corresponding to (10.9) is shown in Fig. 5. It shows that the derivative estimated from the $[u(t + 1) - u(t - 1)]/2$ formula is very accurate (response near 1.0) for low-frequency waves, but that the estimate is rather poor for very high-frequency oscillations (wavelengths shorter than 5 grid spaces).

A better finite difference approximation to a continuous derivative can be devised by using more filter weights. Again, the disadvantage is that more data are lost at the beginning and end of the filtered time series. Using $M = 3$, the following formula can be obtained (Section 7.2 of [3]):

$$y(t) = [u(t + 3) - 9 \cdot u(t + 2) + 45 \cdot u(t + 1) - 45 \cdot u(t - 1)$$
$$+ 9 \cdot u(t - 2) - u(t - 3)]/60$$

Here the ratio of the calculated–true response is given by

$$\frac{\text{calculated}}{\text{true}} = \frac{45 \cdot \sin \omega_j - 9 \cdot \sin(2\omega_j) + \sin(3\omega_j)}{30 \cdot \omega_j}$$

This response is shown as the dashed line in Fig. 5. Note the substantial increase in accuracy for wavelengths of 3–5 grid spaces compared to the simpler formula.
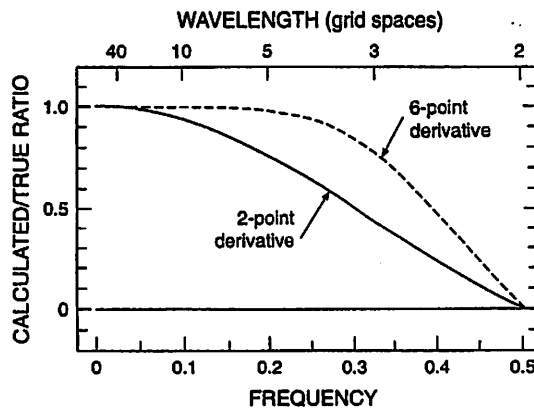

FIG. 5. Calculated vs. true ratio of derivatives estimated using the 2-point and 6-point centered difference operators (Example 3).

10.1.3 Designing a Digital Filter

In this section the technique of designing a specific nonrecursive digital filter is reviewed. This discussion includes low-pass, high-pass, and band-pass filters that have symmetric filter coefficients $(c(-k) = c(k))$, and hence no phase distortion between input and output time series. In the last section, the filter frequency response function was calculated as the finite Fourier transform of the filter coefficients (10.7). In order to derive filter weights $c(k)$ for a desired frequency response function $\bar{c}(\omega_j)$, the inverse transform is used (written here in integral form):

$$c(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \bar{c}(\omega) \cdot e^{i\omega k} \, d\omega \tag{10.10}$$

Now since $c(-k) = c(k)$, only the cosine terms are included in the expansion:

$$c(k) = \frac{1}{\pi} \int_{0}^{\pi} \bar{c}(\omega) \cdot \cos(\omega k) d\omega \tag{10.11}$$

Ideally, one would need an infinite number of coefficients $c(k)$ to exactly match an arbitrary frequency response function $\bar{c}(\omega_j)$. In practice, the number of terms must be truncated to a finite length M. This truncation produces a rippling effect in the frequency response function called the *Gibbs phenomenon*, and such ripples are a generally undesirable feature for digital filters (the nonzero frequency response of the 1-1-1 moving average filter near $f = 0.4$–0.5 (Fig. 1) are an example of such ripples). These ripples in the response function can be reduced by applying a set of weights $w(k)$ to the filter coefficients $c(k)$ to produce a new set of coefficients $c'(k)$; i.e.,

$$c'(k) = c(k) \cdot w(k)$$

These filter weights $w(k)$ are sometimes termed a *window* through which one "sees" the filter weights $c(k)$. The process of truncation of the infinite series $c(k)$ to finite length M can be viewed as using a rectangular window $(w(k) = 1$ for $k = 0$–M, and $w(k) = 0$ otherwise). The practical choice of weighting coefficients or "window shape" depends on several factors and is discussed in detail in Chapter 5 of [1].

One commonly used window or set of weight coefficients with the nice property that it removes a large fraction of the ripple effect is termed the Lanczos window:

$$\begin{aligned} w(k) &= \frac{\sin(\pi k / M)}{(\pi k / M)} & k < M \\ &= 0 & k \geq M \end{aligned} \tag{10.12}$$

This window is used in the calculations that follow. Other choices of windows may be more suited to particular problems, such as the Kaiser window ([1], Chapter 9), which produces sharper transition regions than the Lanczos window, but also results in larger frequency response ripples.

A simple FORTRAN subroutine is included here for generation of windowed filter weights for low-pass, high-pass and bandpass filters. These filters approximate idealized rectangular cutoff frequency response functions with bounds at f_{low} and f_{high}; i.e.,

$$\bar{c}(\omega) = 1 \qquad f_{low} \le \frac{\omega}{2\pi} \le f_{high}$$
$$\qquad\qquad = 0 \qquad \text{otherwise} \tag{10.13}$$

Note that for low-pass filters $f_{low} = 0.0$, and for high-pass filters $f_{high} = 0.5$. The untruncated Fourier expansion for $c(k)$ (10.11) can be evaluated using (10.13):

$$c(k) = \frac{1}{\pi} \int_{2\pi f_{low}}^{2\pi f_{high}} \cos(\omega k)d\omega$$

The unweighted coefficients are then expressed as

$$c(0) = 2(f_{high} - f_{low})$$

$$c(k) = \frac{1}{\pi k} [\sin(2\pi k f_{high}) - \sin(2\pi k f_{low})] \tag{10.14}$$

Finally, these coefficients (10.14) are weighted by the appropriate Lanczos window weights (10.12) to give the final filter coefficients $c'(k)$.

```
      subroutine makefilt (flow, fhigh, nterms, coeffs)
c
c    design nonrecursive bandpass digital filter using Lanczos
c       window
c    → output a symmetric set of filter weights
c
c
c    input:
c              flow - low frequency cutoff (0.0 for a low pass
c                     filter)
c              fhigh - high frequency cutoff (0.5 for high pass
c                      filter)
c                         (frequency units are inverse time or
c                          grid space intervals)
```

```
c               nterms - resulting digital filter length is
c                       (2*nterms+1)
c
c                 → note that the Lanczos weight factor for k= +/-
c                     nterms is zero, so that the filter truncation
c                     point is actually (nterms-1)
c    output:
c               coeffs - digital filter coeffecients
c
c
c    * the maximum (nterms) here is 100, but this can be easily
c      modified
c
      dimension coeffs(-100:100),wt(-100:100)
      do 101 k=-100,100
      coeffs (k)=0.
      wt (k)=0.
  101 continue
c
c     calculate unweighted coefficients
c
      coeffs (0)=2.*(fhigh-flow)
      do 201 k=1,nterms
      tpkfh=2.*3.1415926*k*fhigh
      tpkfl=2.*3.1415926*k*flow
      coeffs(k) = (sin(tpkfh)-sin(tpkfl))/(k*3.1415926)
  201 continue
c
c     now calculate Lanczos weights (sigma factors)
c        (or can substitute a different window here
c        such as the von Hann or Kaiser windows)
c
      wt(0)=1.0
      do 301 k=1,nterms-1
      phi=3.1415926*float(k)/float(nterms)
      wt(k)=sin(phi)/phi
  301 continue
c
c     weighted filter coefficients
c
      do 401 k=0,nterms
      coeffs(k)=coeffs(k)*wt(k)
```

```
  401 continue
c
c    symmetric filter weights
c
      do 403 k=1,nterms
      coeffs(-k)=coeffs(k)
  403 continue
      return
      end
```

This subroutine generated the set of coefficients used in Example 2 by setting $f_{low} = 0.0$, $f_{high} = 0.13$, and $M = 5$ (the Lanczos weights are 0 for the $k = M$ term, so that the filter is effectively truncated at $k = M - 1$).

Example 4: As one further example, a bandpass filter is designed to specifically isolate the 40–50 day oscillations in the Canton Island wind time series. The idealized rectangular frequency response function is chosen such that $f_{low} = 0.013$ and $f_{high} = 0.031$ (ideally retaining periods in the range 32–77 days). This is a very narrow spectral band and requires a large number of filter coefficients for an accurate approximation; here $M = 60$ is chosen (so that 60 days of data are lost from each end of the time series). The frequency response of this filter is shown in Fig. 6 (note that the
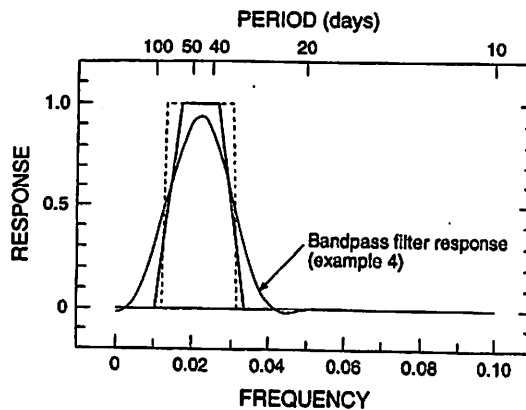

FIG. 6. Smooth continuous line shows frequency response function for bandpass digital filter of Example 4; dashed lines show idealized rectangular response with frequency limits $f_{low} = 0.013$ and $f_{high} = 0.031$. The trapezoid-shaped curve shows the frequency response function used in the direct Fourier analysis–resynthesis calculation of Section 10.1.4. Note the abscissa in this figure only covers frequencies 0.0 to 0.10.

frequency range plotted in Fig. 6 is only $f = 0.0$–0.1). The resulting filtered time series is shown as a solid line in Fig. 7, isolating the 40–50 day period oscillations. Note the amplitude modulation of the 40–50 day oscillation in time, with maximum amplitudes over January–February for both years.

10.1.4 Filtering via Direct Fourier Analysis–Resynthesis

An alternative method for filtering time series is to use a direct Fourier analysis to obtain frequency coefficients, truncate the coefficients in such a manner so that only a specified frequency band is retained, and then resynthesize the time series using the truncated coefficients. This method is easily implemented by the use of direct and inverse fast Fourier transforms (FFTs), which are now popular and readily available. As an example, this procedure was used to bandpass filter the wind time series over a frequency range similar to that chosen in Example 4. First, an FFT of the entire data is used to generate frequency coefficients. Second, the coefficients are weighted with the trapezoid-shaped frequency response function shown as the heavy curve in Fig. 6; i.e., frequency coefficients outside this band are set to 0. A trapezoid shape is chosen for the frequency window, as opposed to a rectangular shape, because too sharp of a frequency cutoff will result in a rippling effect in the filtered time series, an effect called *ringing*. Finally, the windowed coefficients are used to synthesize the filtered time series. The result is included as a dashed line in Fig. 7. Note the similarity to the digital filtered data; the FFT and digital filter results would be *exactly* the
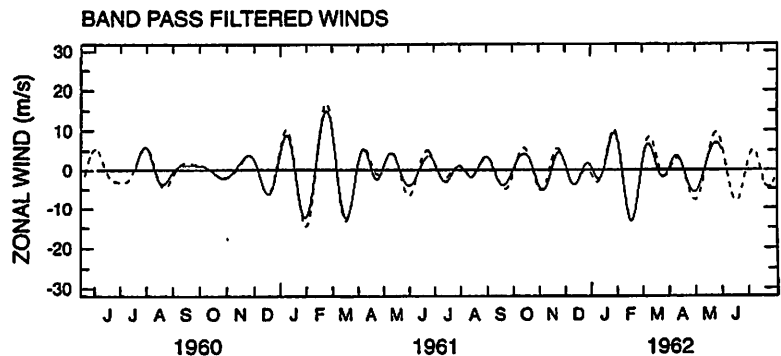
BAND PASS FILTERED WINDS

FIG. 7. Time series of Canton Island winds (Fig. 2) after application of 40–50 day bandpass digital filter (Example 4, solid line) and Fourier analysis–resynthesis (Section 10.1.4, dashed line).

same if the FFT window (Fig. 6) had the same shape as the bandpass filter response function.

At first inspection it appears that the FFT filtering method is superior to the digital filtering in that data are not lost at the beginning and end of the time series (see Fig. 7). However, the endpoint data generated by the FFT method are not to be believed (and should not even be displayed in plots of filtered data). The reason for this is that the FFT analysis implicitly assumes that the data are perfectly periodic with repeat distance N (the length of the time series), so that the beginning and end of the time series are implicitly tied together (see the example later in Fig. 11). Hence the FFT bandpass filtered data near the beginning and end of the data are contaminated from the "other" end. This effect is clearly seen upon careful inspection of the beginning and end sections of the original (Fig. 2) versus bandpass filtered data (Fig. 7). The original data show maxima of opposite signs near the ends, whereas the FFT bandpassed data show a "turning over" of the curves near the ends, clearly not tracking the original data.

10.2 Data Preprocessing for Spectral Analysis

Spectral analysis is the name given to estimating the power spectral density function (10.4) (or cross-spectral density) from time series of observed data. Spectral analysis is useful in analyzing time series because it allows for a rearranging of the data according to frequency rather than time sequence. This is often useful in physical sciences because many phenomena are naturally separated by their frequency characteristics (such as high-frequency day-to-day weather variations versus low-frequency seasonal changes). Considerations regarding the proper calculation and significance of power and cross-spectral quantities are discussed in Chapter 11. In this section the discussion focuses on two topics that are of practical concern for spectral analysis: data windowing (tapering) and the removal of background trends prior to spectral analysis. As a note, trend removal should be done first and tapering second, prior to spectral analysis; they are discussed in opposite order here because the concepts of data windowing and leakage are central for understanding why trend removal is important.

10.2.1 Data Windowing (Tapering)

One fundamental problem in the proper estimation of spectral quantities from real data occurs because the time series analyzed are of finite length. Hence estimates of the "true" frequency spectrum of some variable are made based on a finite length sample of that variable (a measured time
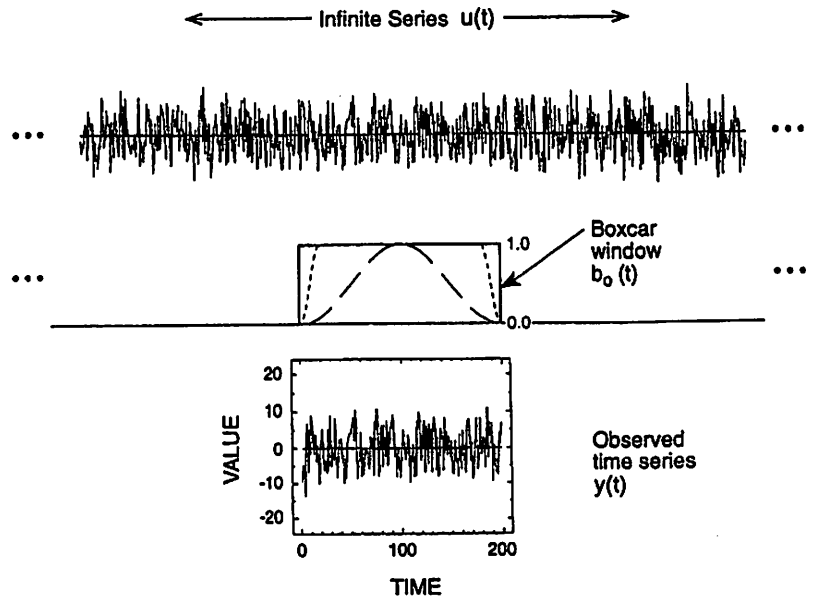
FIG. 8. Schematic representation of a finite length time series (bottom) as an infinite time series (top), sampled with a rectangular "boxcar" window (heavy curve in middle). Also shown in the middle are the Hanning and cosine-taper windows (long and short dashes, respectively.).

series). This situation is shown graphically in Fig. 8. The observed time series $y(t)$ can be viewed mathematically as an infinite time series $u(t)$ multiplied by a "boxcar" function $b_0(t)$ (where $b_0(t) = 1$ for $t = -M, \ldots,$ M and $b_0(t) = 0$ otherwise; the data length N is thus $2M + 1$):

$$y(t) = b_0(t) \cdot u(t)$$

Now the discrete Fourier transform (10.2) of the observed series $y(t)$ may be written as

$$\bar{y}(\omega_j) = \sum_{t=-M}^{M} y(t) \cdot e^{-i\omega_j t}$$

$$= \sum_{t=-\infty}^{\infty} b_0(t) \cdot u(t) \cdot e^{-i\omega_j t}$$

$$= \sum_{t=-\infty}^{\infty} b_0(t) \cdot e^{-i\omega_j t} \sum_{l=-\infty}^{\infty} \bar{u}(\omega_l) \cdot e^{i\omega_l t} \qquad (10.15)$$

$$= \sum_{l=-\infty}^{\infty} \tilde{u}(\omega_l) \cdot \sum_{t=-\infty}^{\infty} b_0(t) \cdot e^{-i(\omega_j - \omega_l)t}$$

$$= \sum_{l=-\infty}^{\infty} \tilde{u}(\omega_l) \cdot \tilde{b}_0(\omega_j - \omega_l)$$

This shows that the frequency coefficients calculated from the observed time series are equivalent to the "true" coefficients from an infinite time series $\tilde{u}(\omega_l)$, convoluted with the transform of the boxcar function $\tilde{b}_0(\omega_j - \omega_l)$. In other words, each calculated spectral estimate $\tilde{y}(\omega_j)$ is a sum of the "true" spectral estimates near ω_j, weighted with this function $\tilde{b}_0(\omega_j - \omega_l)$. From the definition of $b_0(t)$, $\tilde{b}_0(\omega_j - \omega_l)$ is easily evaluated:

$$\tilde{b}_0(\omega_j - \omega_l) = \sum_{t=-\infty}^{\infty} b_0(t) \cdot e^{-i(\omega_j - \omega_l)t}$$

$$= \sum_{t=-M}^{M} e^{-i(\omega_j - \omega_l)t} \tag{10.16}$$

$$= \frac{2 \cdot \sin(\omega_j - \omega_l) \cdot M}{(\omega_j - \omega_l)}$$

This function is shown in Fig. 9. The finite width of the central maximum in $\tilde{b}_0(\omega_j - \omega_l)$ (called the *bandwidth* of the spectral analysis, because it sets
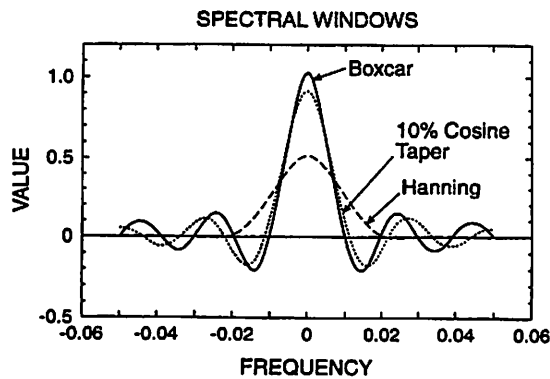

FIG. 9. Frequency transforms of the boxcar, Hanning, and cosine-taper windows (shown in the middle of Fig. 8), calculated as in (10.16), using $M = 100$ (i.e., for a time series of length 200).

a limit on the frequency resolution), and the presence of ripples, results in smearing and mixing up the "true" frequency components. This general problem is termed *leakage*, in that the true spectral coefficients are transferred or "leak" between adjacent and distant frequency bands. Note that, as the length N of the time series increases, the transform function $\tilde{b}_0(\omega_j - \omega_l)$ (10.16) gets narrower, and hence better estimates of the "true" spectrum are made for longer time series. However, this problem remains to some degree for all finite length time series.

Several methods are used in practice in an attempt to minimize the leakage due to the "boxcar" sampling and associated transform $\tilde{b}_0(\omega_j - \omega_l)$. These methods are all similar in that they choose a different "window" than the boxcar function, with the aim of reducing the ripples in the transform $\tilde{b}_0(\omega_j - \omega_l)$. These windows have smooth transitions to 0 at the beginning and end of the data record, and this general process is termed *tapering* the data. Tapering is applied to a time series by simply multiplying by the chosen window function. One common choice is to use a cosine-shaped window of the form

$$b_1(t) = \frac{1}{2}\left[1 + \cos\left(\frac{\pi t}{M}\right)\right] = \cos^2\left(\frac{\pi t}{2M}\right) \qquad (10.17)$$

This is called a Hanning window and is shown for comparison to the boxcar window in Fig. 8. The transform of this function, $\tilde{b}_1(\omega_j - \omega_l)$, is shown in Fig. 9. Note that there is a reduction in the ripples compared to the boxcar transform (and hence a reduction in leakage from "distant" frequencies), but also that the bandwidth for the Hanning window transform is approximately twice as wide (i.e., the resulting spectrum will have less frequency resolution). Also, the Hanning window uses only the central half or so of the data record and will give a bit different results if the signals are not statistically stationary (i.e., the same throughout the record). One other frequently used window is a rectangular or boxcar function, modified so that the beginning and end sections have a smooth cosine-tapered structure. For example, the first and last 10% of the data may be tapered according to

$$
\begin{aligned}
b_2(t) &= \frac{1}{2}\left[1 + \cos\left(\frac{5\pi t}{M}\right)\right] & -M &\le t \le -\frac{4}{5}M \\
&= 1 & -\frac{4}{5}M &< t < \frac{4}{5}M \\
&= \frac{1}{2}\left[1 + \cos\left(\frac{5\pi t}{M}\right)\right] & \frac{4}{5}M &\le t \le M
\end{aligned}
$$

This window and its transform are shown in Figs. 8 and 9, respectively. This "10% cosine taper" window provides only a slight reduction in leakage compared to the boxcar window.

The inherent problem of leakage is the reason why harmonic amplitude estimates for individual frequency coefficients are biased and possess large uncertainties, even for very long time series; this is why the periodogram (10.4) is *not* a good estimator for the spectral power density. As discussed in more detail in Chapter 11, one method to reduce both the bias and variance of spectral power estimates is to average the periodogram over the individual frequency bands. This can be done in one of two ways: (1) averaging several different realizations of the spectra, i.e., average the power spectra from several time series, or (2) averaging a single spectrum over several adjacent frequency bands (i.e., applying a smoothing filter to the periodogram). For the latter case, the character of the resulting spectrum depends on the degree of smoothing: averaging over few frequency bands (narrow bandwidth) will result in many peaks in the spectrum, some of which may be spurious, while a wider bandwidth may smooth the spectrum so much that no peaks are distinguishable. In practice, the type of smoothing needs to be determined for each problem at hand, usually by testing several variations; further examples may be found in [4], [5], and [6].

An example of spectral smoothing is shown in Fig. 10, based on analysis of a thousand point sample of the synthetic time series shown in Fig. 8. This time series was generated by choosing two sharp spectral peaks at frequencies $f = 0.05$ and $f = 0.35$ and adding a component of random number generated noise to the time series (the latter adds variance to the spectrum at all frequencies, so-called white noise). Figure 10(a) shows the sample power spectrum obtained by applying a Hanning window (10.17) to the data, calculating the harmonic coefficients using a fast Fourier transform, and calculating the spectrum from (10.4). Note the relatively narrow bandwidth associated with this analysis and the large variability between adjacent frequency bands. Figure 10(b) shows the spectrum obtained using no data tapering (i.e., the boxcar window), but with the periodogram power estimates smoothed in frequency using a moving Gaussian-shaped filter (whose width if indicated as BW in Fig. 10(b)). Note the much smoother character of this spectrum compared to that in Fig. 10(a), although in this example the spectral peaks at $f = 0.05$ and 0.35 are clearly evident in both calculations. One note regarding the calculations in Fig. 10: because the Hanning window tapers the ends of the data, the time series variance (and spectral density estimates) are reduced compared to that for the full time series. For direct comparisons here, the Hanning window spectral power estimates have been multiplied by a constant factor, so that the sums over frequency of the two spectral estimates in Fig. 10 are equal.
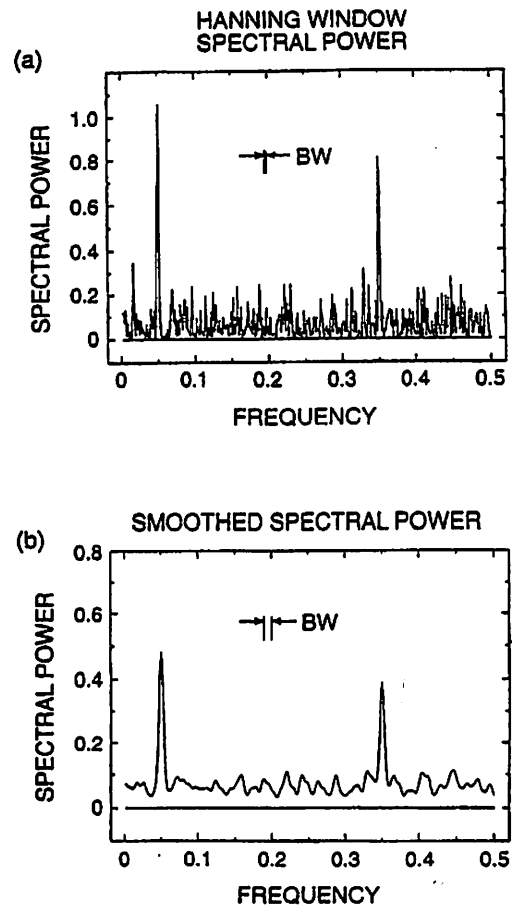
FIG. 10. Sample spectral power densities calculated from 1000-point records of the time series shown in Fig. 8. Spectra in (a) are the periodogram of the data using a Hanning data window, while the estimates in (b) result from frequency smoothing the periodogram derived from untapered data. BW denotes the bandwidth of the analyses.

10.2.2 Removal of Background Trends

Another potential problem in spectral analysis of finite length time series due to the presence of strong background signals or trends over the length of the data record. The fundamental problem is that the background signal contributes to additional background frequency components, which in turn get mixed with the "true" coefficients via the

leakage arguments discussed previously. One way to visualize the problem is to note that a finite length Fourier analysis of time series $u(t)$ ($t = 0, 1, 2, \ldots, N - 1$) implicitly assumes that the data are perfectly periodic with a repeat distance N. This is shown schematically in Fig. 11, using for an example a 30-yr time series of atmospheric carbon dioxide (CO_2) measurements taken over Hawaii, obtained from [7]. There is a clear increase in time (positive trend) in these data that is a result of the systematic build-up of carbon dioxide in the atmosphere due to burning fossil fuels; there is also a strong annual cycle related to seasonal plant growth in the Northern Hemisphere. The result of the implicit periodic nature of the finite Fourier transform is that the time series effectively has the overall shape of a "sawtooth" curve, with strong discontinuities every 30 yrs due to the trend in the data. Power spectra calculated from this time series will have strong amplitudes over a broad frequency range introduced solely as a result of this trend (and mixed into the "true" spectrum by the leakage arguments discussed earlier). Figure 12 compares power spectrum estimates calculated from this time series, with and without the background trend removed. The trend was
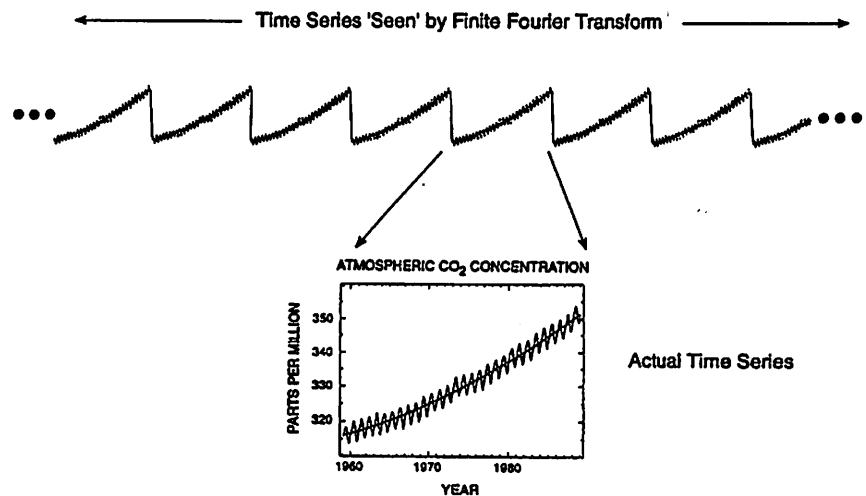

FIG. 11. Top curve shows a schematic representation of the infinitely periodic time series "seen" by a finite Fourier transform of the 30-yr record below. The time series is atmospheric carbon dioxide concentration (in parts per million by volume) measured at Launa Loa, Hawaii, over 1959–1988 [7]. Note that the strong trend (denoted by the smoothed curve in the lower figure) introduces a "sawtooth" shape to the periodic time series above.
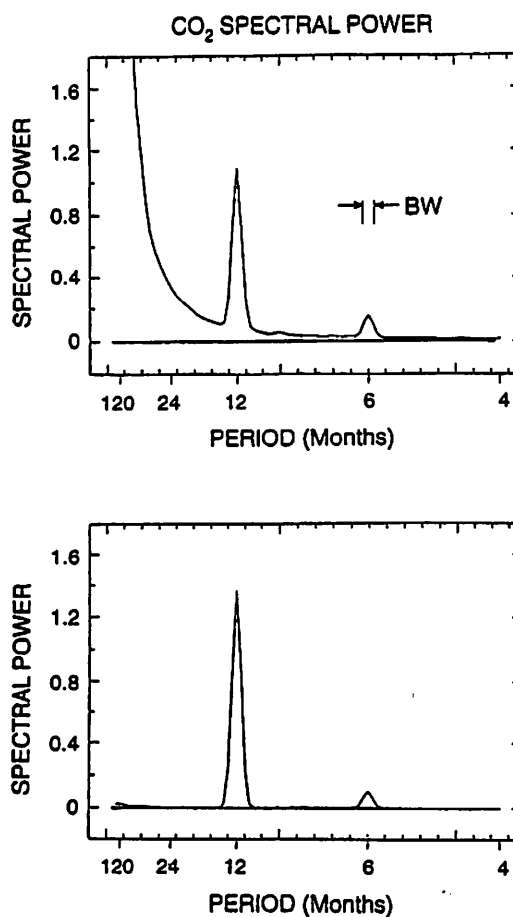
FIG. 12. Spectral power density for the carbon dioxide time series in Fig. 11, calculated before (top) and after (bottom) removal of the background trend (the trend is indicated by the smooth curve in the lower panel of Fig. 11).

estimated with a least squares fit to a quadratic function over the 30 yrs; i.e.,

$$y(t) = a_0 + a_1 t + a_2 t^2$$

with the coefficients a_0, a_1, and a_2 calculated according to the formulas given in Section 8.3 of [8]. This calculated background curve is indicated in the center of Fig. 11. (The choice of using a quadratic function for the background was somewhat arbitrary; other functional forms could have

been chosen, or the data could be high-pass filtered to remove the slow background trend.) Comparison of the power spectra before and after trend removal (Fig. 12) clearly shows the spurious power introduced by the trend in this example: note also the changed magnitude of the annual harmonic between these two calculations and the fact that the detrended data spectrum shows a *larger* peak for the annual cycle. The smaller annual cycle peak in the original data spectrum results from the negative sidelobes associated with the boxcar convolution function in Fig. 9, so that the annual harmonic amplitudes (and resulting spectral power densities) were *decreased* by leakage in this example.

Trend removal is the simplest form of *prewhitening* of data prior to spectral analysis. In general, *prewhitening* refers to some linear transformation of the data in order to get an overall smoother spectrum in frequency space (constant power versus frequency is termed a *white* power spectrum, hence the name *prewhitening*). This preprocessing step can be important to minimize the effects of leakage discussed previously.

10.3 Imperfectly Sampled Time Series

The preceding section discussed biases that occur in spectral analysis because of finite data length and the presence of background trends. One other subject discussed here is how to handle time series with missing data values or time series that have unequally spaced observations. A simple technique is shown here for the estimation of power spectra from such data. There is one further general problem that occurs as a result of undersampling the true variability in a time series; undersampling in this context means not sampling frequently enough. This undersampling results in a misrepresentation of the true frequency dependence of the spectrum, an effect known as *aliasing*. An example of aliasing is included to illustrate this effect.

10.3.1 Calculating Power Spectra for Time Series with Missing Data

Time series may have missing data points for several reasons. Observations may simply be unavailable for certain time periods; this is a frequent problem in historical records of meteorological data, for example. Data values that are clearly spurious may also be present in a data record due to some detector malfunction. Removal of these wild points (or outliers) is important prior to subsequent spectral analyses, or else they will contaminate the entire spectrum. There are several ways to identify such outliers; often a simple plot of the time series will reveal obviously bad data points. Statistical methods may also be used to identify outliers. For example, the

standard deviation of the entire data record can be calculated, and data that are more than (say) three standard deviations from the mean can be omitted. In any case, we wish to consider how to calculate spectra from time series with such missing values.

There are three straightforward methods to calculate spectra from such time series. First, the missing data can be interpolated in some manner (using linear interpolation or some more complicated scheme) and then spectral analysis performed on the resulting "complete" time series. This is the easiest solution for a relatively small amount of missing data. The problem with this approach in general is that structure is built into the resulting power spectrum by the type of interpolation used (any interpolation scheme can be thought of as a filter, with a specified frequency response, see [1], Section 3.7; this frequency response is then partially mirrored in the resulting power spectrum).

Second, discrete Fourier transform coefficients can be estimated from the time series by the least-squares fitting of the data to the individual harmonics, one at a time. Equations for these calculations can be found in Section 2.2 of [9].

A third technique is based on calculating the power spectrum using the lag-correlation technique (e.g., Section 7.4 of [4]). This calculation is based on the fact that the power spectrum is equivalent to the finite Fourier transform of the lag autocovariance function of a time series. This technique was a commonly used method of spectral analysis prior to the introduction of fast Fourier transforms. Briefly the calculation is as follows:

a. Calculate the lag autocovariance function $C(\tau)$ up to some finite maximum lag T:

$$C(\tau) = \frac{1}{N} \sum_{t=0}^{N-1-\tau} u(t) \cdot u(t + \tau) \tag{10.18}$$

where the time mean values (and trends) have been removed from the series $u(t)$. Note that some authors choose a normalization factor of $1/(N - \tau)$ instead of the $1/N$ in front of the definition of $C(\tau)$, e.g., Bath ([5], Section 3.3.3). Jenkins and Watts ([10], Section 5.3.3) discuss these two options, choosing the $1/N$ expression because it has a smaller mean squared error.

b. Calculate the (unsmoothed) power spectrum by the finite Fourier transform of $C(\tau)$. Because $C(\tau)$ is symmetric in τ, only the cosine terms survive:

$$Y(\omega_j) = \frac{1}{T} \left[C(0) + 2 \sum_{\tau=1}^{T-1} C(\tau) \cdot \cos(\omega_j \tau) + C(T)(-1)^\tau \right] \tag{10.19}$$

c. The resulting spectrum is smoothed in some manner; a common choice is to use a running .25–.50–.25 smoothing in frequency (see Section 7.4 of [4]).

The maximum number of lags (T) is chosen based on the length of the time series and the desired spectral resolution; it is inversely proportional to the bandwidth of the analysis (larger T results in higher spectral resolution but less statistical stability). In practice the user needs to choose T and the method of smoothing the spectrum that balances resolution versus stability.

This calculational procedure is directly applicable to time series with missing data by simply ignoring the missing data in step (a); i.e., only the available data are used in calculating $C(\tau)$:

$$C(\tau) = \frac{1}{L} \sum_{t=0}^{L-1-\tau} u(t) \cdot u(t + \tau) \tag{10.20}$$

with L being the amount of data pairs that are available at each respective time lag. A recent example of application of this method to the spectral analysis of satellite ozone data with missing observations is found in [11].

We test this analysis technique here by analyzing the same synthetic time series (of length 1000) used in Section 10.2.1. The power spectrum calculated from this time series by the lag correlation technique with maximum lag $T =$ 500 is shown in Fig. 13. Note that this spectrum is nearly identical to that produced by the smoothed periodogram estimate shown in Fig. 10(b); this
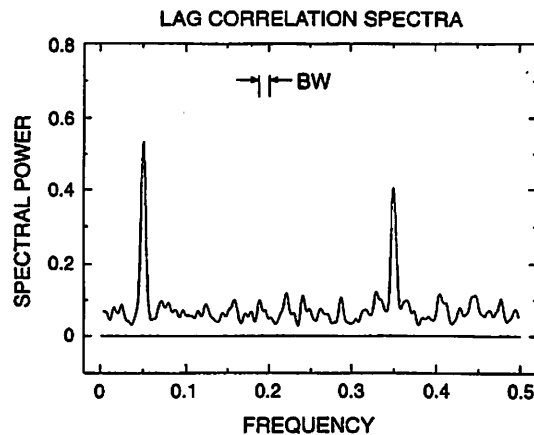

FIG. 13. Power spectral density calculated from 1000 day time sample of the time series shown in Fig. 8, calculated using the lag-correlation analysis technique (Section 10.3.1).

similarity is due to choosing the maximum lag (in 10.19) to be $T = 500$, resulting in a bandwidth similar to that used in the periodogram smoothing. The effect of missing data on these spectral estimates is tested by removing some percentage of the data from the time series in a random manner and recalculating the power spectrum. Figure 14 shows power spectra calculated from the data with 10%, 30%, and 50% of the data randomly removed (compare this to the "original" spectrum in Fig. 13). For 10% and 30% of the data missing the spectra are quite similar to the original, with a slight reduction in intensity of the peaks and increase in the background "noise" level. The peaks
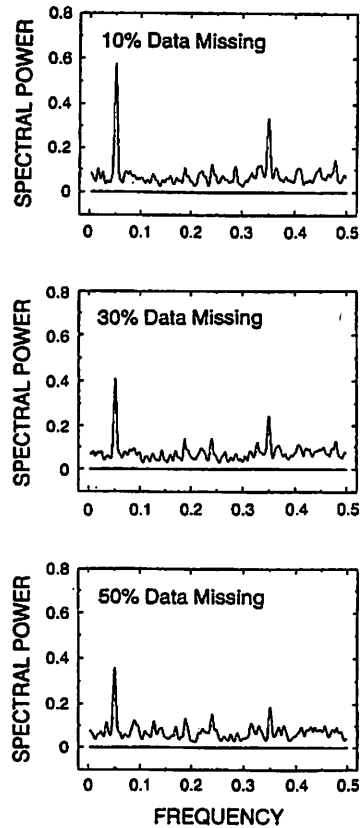

FIG. 14. Power spectral density estimates for the same time series analyzed in Fig. 13, but with successively larger amounts of the data removed prior to calculation of (10.20). Shown are spectra for 10% of the data missing (top), 30% and 50% (bottom). The spectrum for the full time series (no data missing) is shown in Fig. 13.

are reduced and the background increased further for the case with 50% of the data missing, so that the peak at $f = 0.35$ cannot be distinguished from other spurious maxima. If much more than 50% of the data is removed in this example, the larger peak at $f = 0.05$ also becomes indistinguishable, showing that this is near the limit where useful spectra can be calculated in this case. Note that the sharp, well-defined spectral peaks chosen for this example (Fig. 13) make it a highly idealized case and that if the peaks were smaller they would not stand out against the background for the 50% missing data case. In general the amount of missing data that can be tolerated depends on the character of the data being analyzed. For analyses of real time series, it is recommended that tests similar to this one be done to determine the effects of variable missing data.

10.3.2 Aliasing

One additional important problem in spectral analysis arises due to the presence of variability with frequencies higher than those that can be resolved by the sampling rate. For a given data spacing Δt, the highest frequency oscillation that can be resolved is one with period $2\Delta t$, or frequency $f = 1/(2\Delta t)$. This high-frequency limit is called the *Nyquist frequency* f_N. In the case where there is substantial variability at frequencies above f_N, the effect of sampling with spacing Δt is that power at frequencies above f_N will appear as power at frequencies lower than f_N. This effect is called *aliasing*. Specifically, power at frequencies f, $2f_N \pm f$, $4f_N \pm f$, . . . are all aliased into (appear as) power at frequency f, due solely to the sampling rate Δt.

An example is shown here based on the synthetic time series analyzed previously, whose spectrum is shown in Figs. 10 and 13 (based on sampling at every time step, $\Delta t = 1$). This same data is sampled at every second point instead of at every point, so that $\Delta t = 2$ versus $\Delta t = 1$ previously. The Nyquist frequency for the $2\Delta t$ sampled data is $f_N = 1/(2 * 2\Delta t) = 0.25$. The original power spectrum (Fig. 13) showed that there is a spectral peak in the original data at $f = 0.35$, above this new Nyquist limit, so that this power will be aliased in the new sampling at $\Delta t = 2$. The frequency where this aliased power will occur is at $f_a = 2 f_N - f = 2 (0.25) - 0.35 = 0.15$. A power spectrum calculated from the newly sampled data is shown in Fig. 15, and a clear peak is indeed found at $f = 0.15$, due to aliasing. (One way to visualize the effect of this aliasing is that the power in the "true" spectrum (Fig. 13) has been "folded" back about the Nyquist frequency $f_N = 0.25$). Note that if the higher frequency peak in the "original" spectrum (Fig. 13) had been located near $f = 0.45$, it would have aliased onto the preexisting spectral peak at $f = 0.05$, so that the power for that peak would have been severely overestimated.
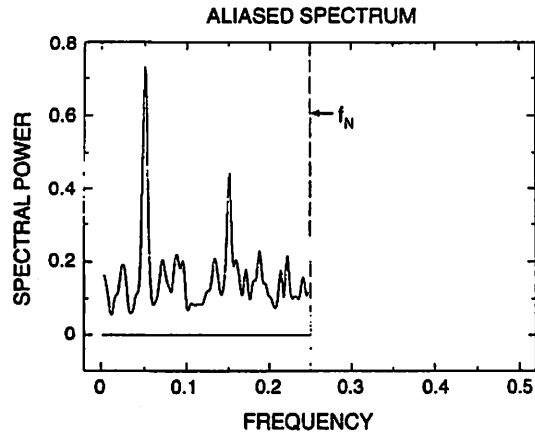
ALIASED SPECTRUM

FIG. 15. Power spectral density calculated from a sample of the time series in Fig. 8, with data sampled at every second point. The Nyquist frequency is $f_N = 0.25$. The spectral peak near $f = 0.15$ results from aliasing of the power in the "true" spectrum near $f = 0.35$ (shown in Fig. 13).

Note that once the data have been sampled at $2\Delta t$, there is nothing to distinguish the peak at $f = 0.15$ as a "false" peak resulting from aliasing; nothing can be done once the data are sampled. The only way to avoid aliasing is to choose a high enough sampling rate (small Δt). This is not always an option in practice, however, and the user should be aware of potential aliasing problems in analysis of any time series.

Acknowledgments

References

1. Hamming, R. W. (1989). *Digital Filters*, 3d ed. Prentice-Hall, Englewood Cliffs, NJ.
2. Madden, R. A., and Julian, P. R. (1971). "Detection of a 40–50 Day Oscillation in the Zonal Wind in the Tropical Pacific." *J. Atmos. Sci.* **28**, 702–708.

3. Maron, M. J. (1982). *Numerical Analysis: A Practical Approach.* Collier Macmillan, New York.

4. Chatfield, C. (1989). *The Analysis of Time Series: An Introduction,* 4th ed. Chapman and Hall, London.

5. Bath, M. (1974). *Spectral Analysis in Geophysics.* Elsevier, Amsterdam.

6. Priestly, M. B. (1981). *Spectral Analysis and Time Series.* Academic Press, London.

7. Boden, T. A., Karciruk, P., and Farrel, M. P. (1990). "Trends '90. A Compendium of Data on Global Change." Carbon Dioxide Information Analysis Center, Oak Ridge, TN.

8. Chatfield, C. (1983). *Statistics for Technology,* 3d ed. Chapman and Hall, London.

9. Bloomfield, P. (1976). *Fourier Analysis of Time Series: An Introduction.* John Wiley and Sons, New York.

10. Jenkins, G. M., and Watts, D. G. (1969). *Spectral Analysis and Its Applications.* Holden-Day, San Francisco.

11. Randel, W. J., and Gille, J. C. (1991). "Kelvin Wave Variability in the Upper Stratosphere Observed in Satellite Ozone Data." *J. Atmos. Sci.* **48,** 2336–2349.